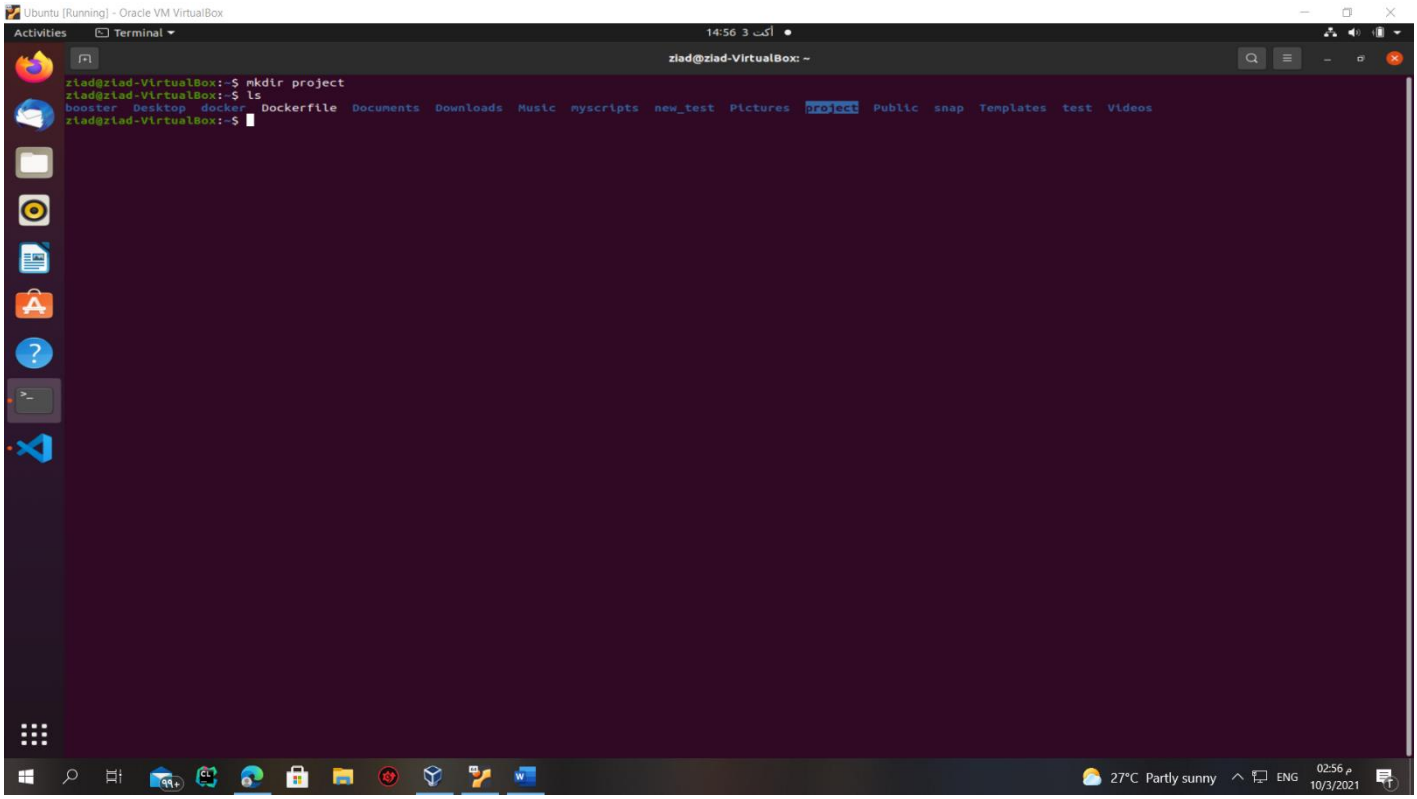
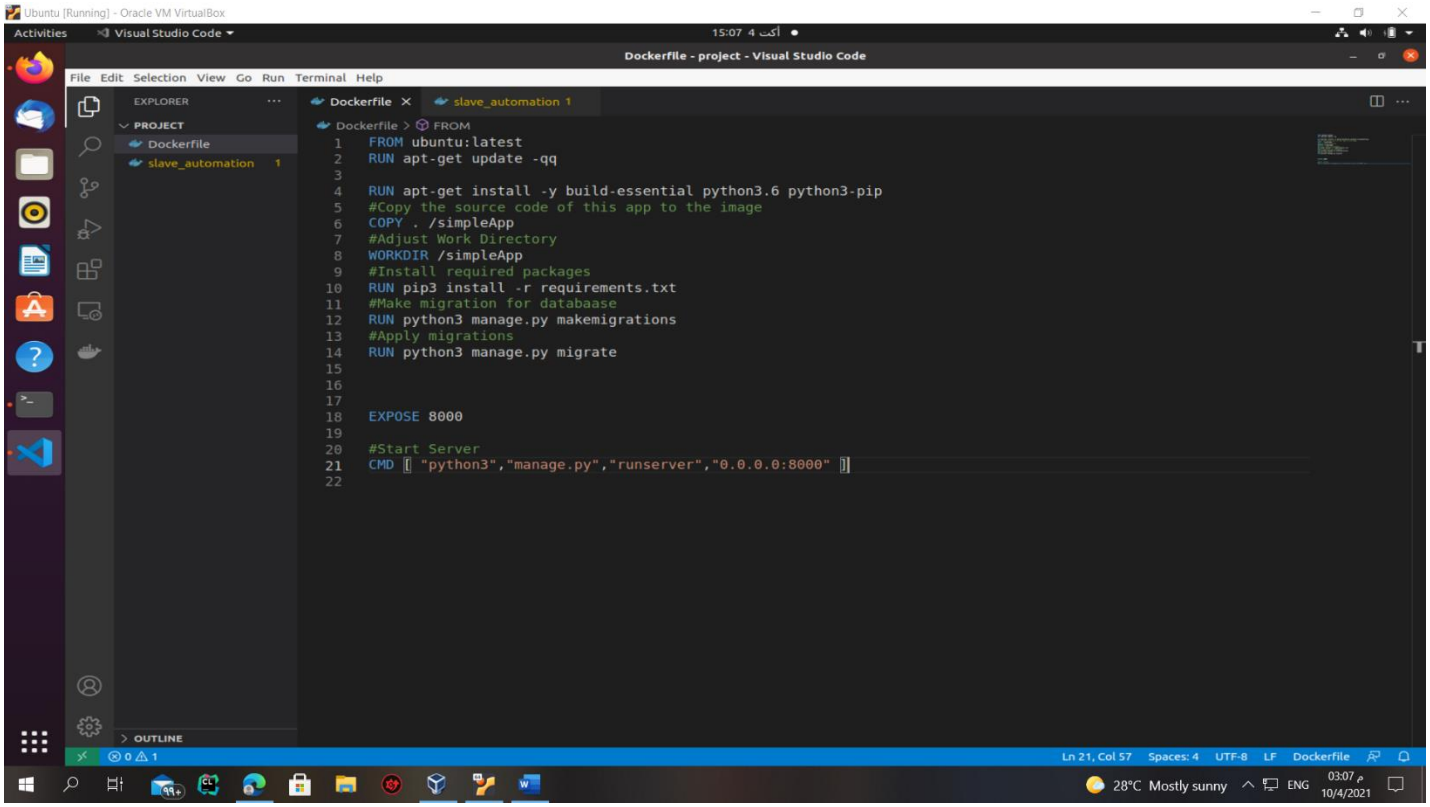
Deploying a Django App: The Process

1-Create A project directory

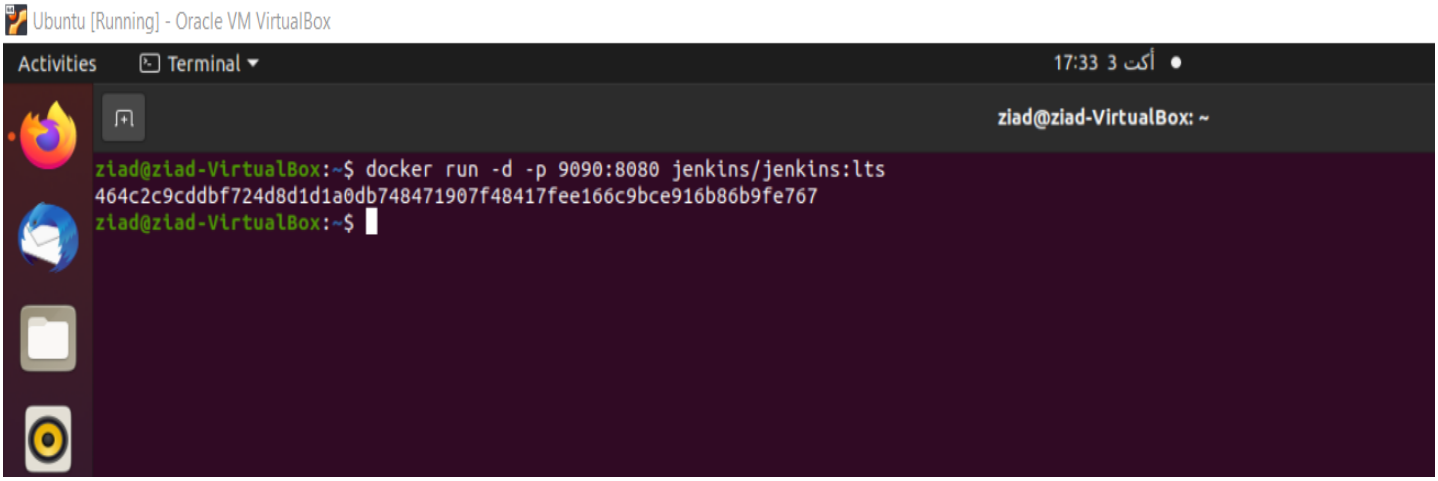


2- Create Docker File for the Django App inside Directory



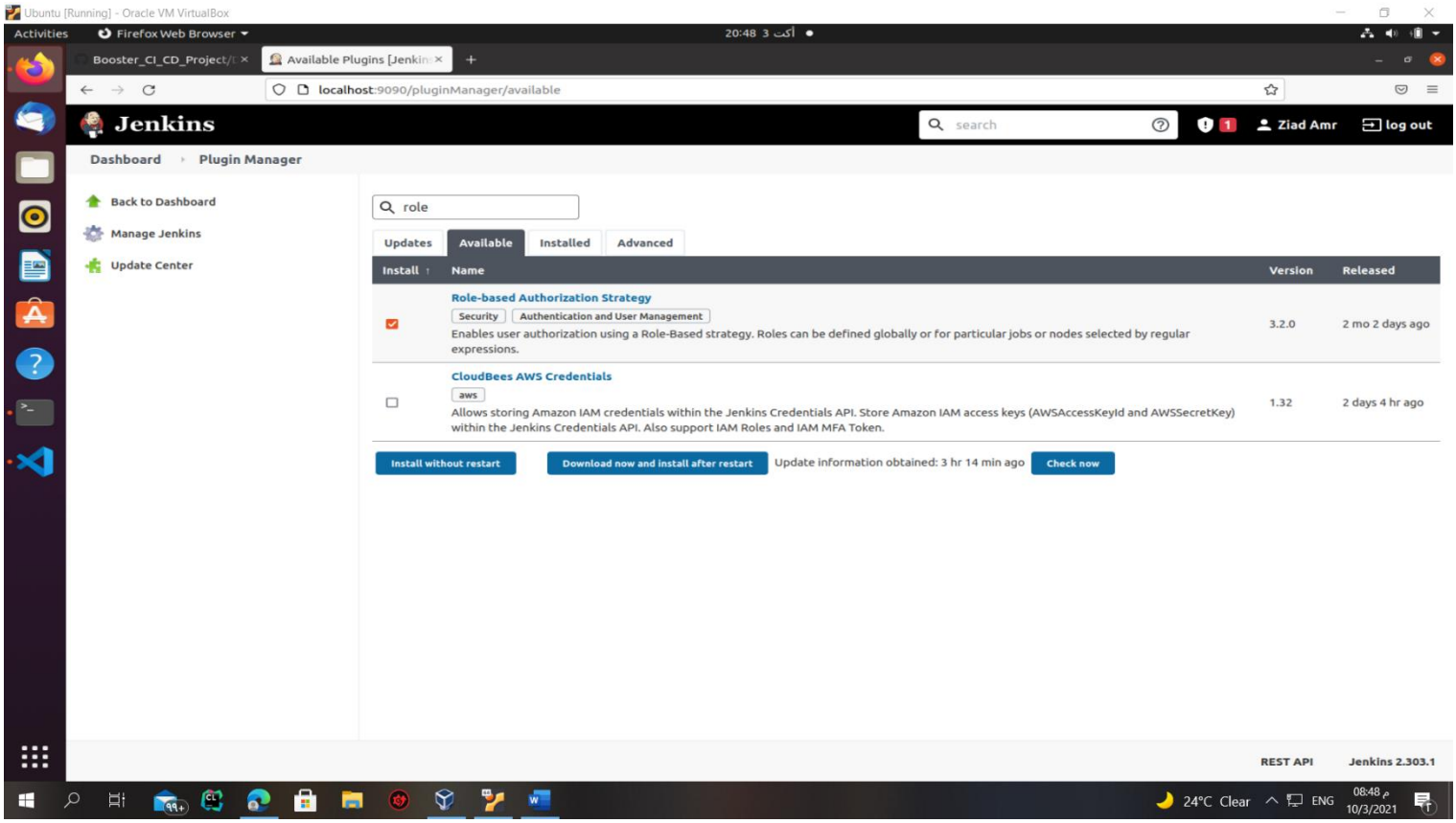
4-Now let’s start a Jenkins Server

a) We will start Jenkins as a docker container on port 9090

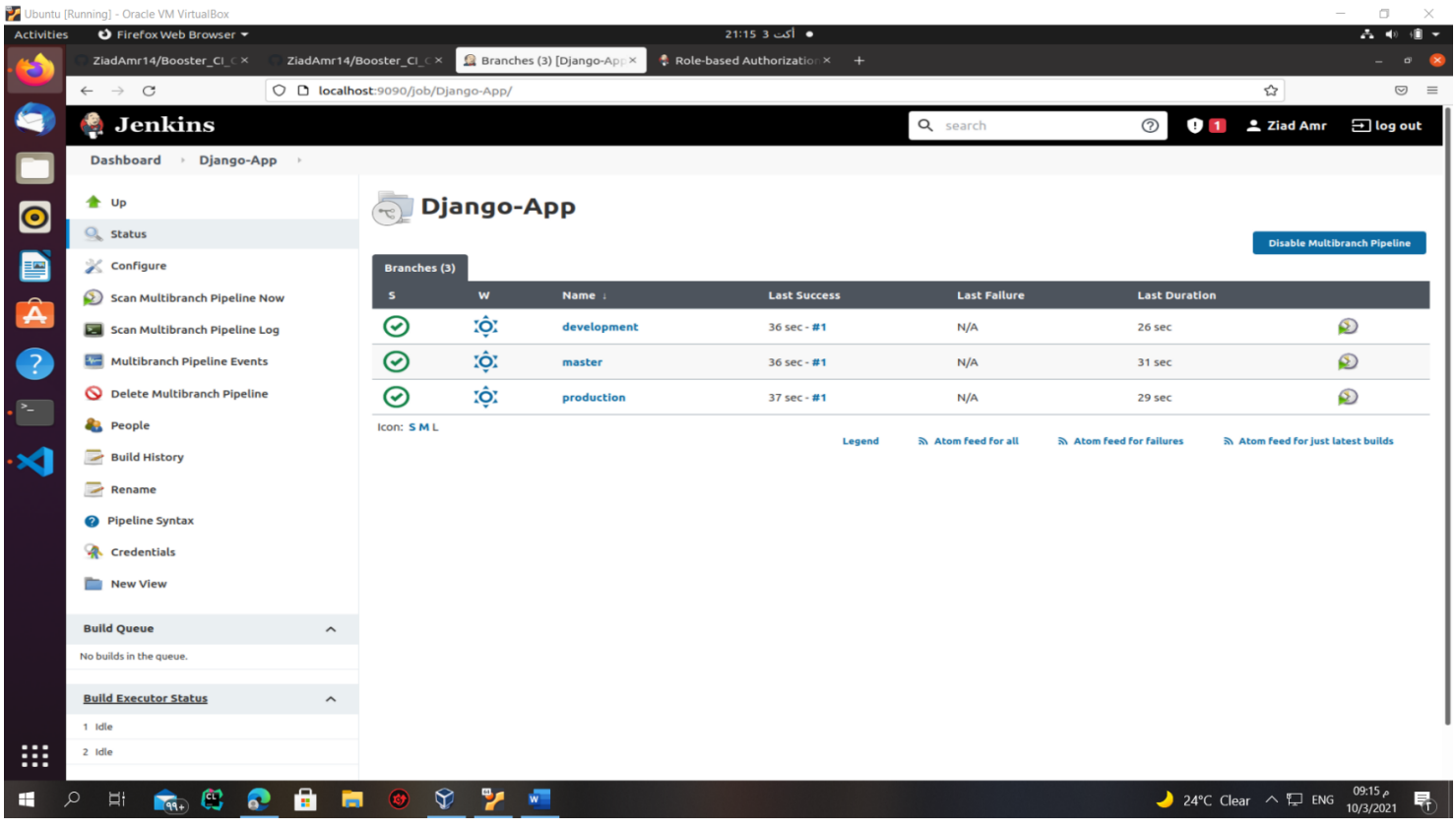


b) Now we configure the Jenkins server

i) Install Role-Based Authorization Strategy

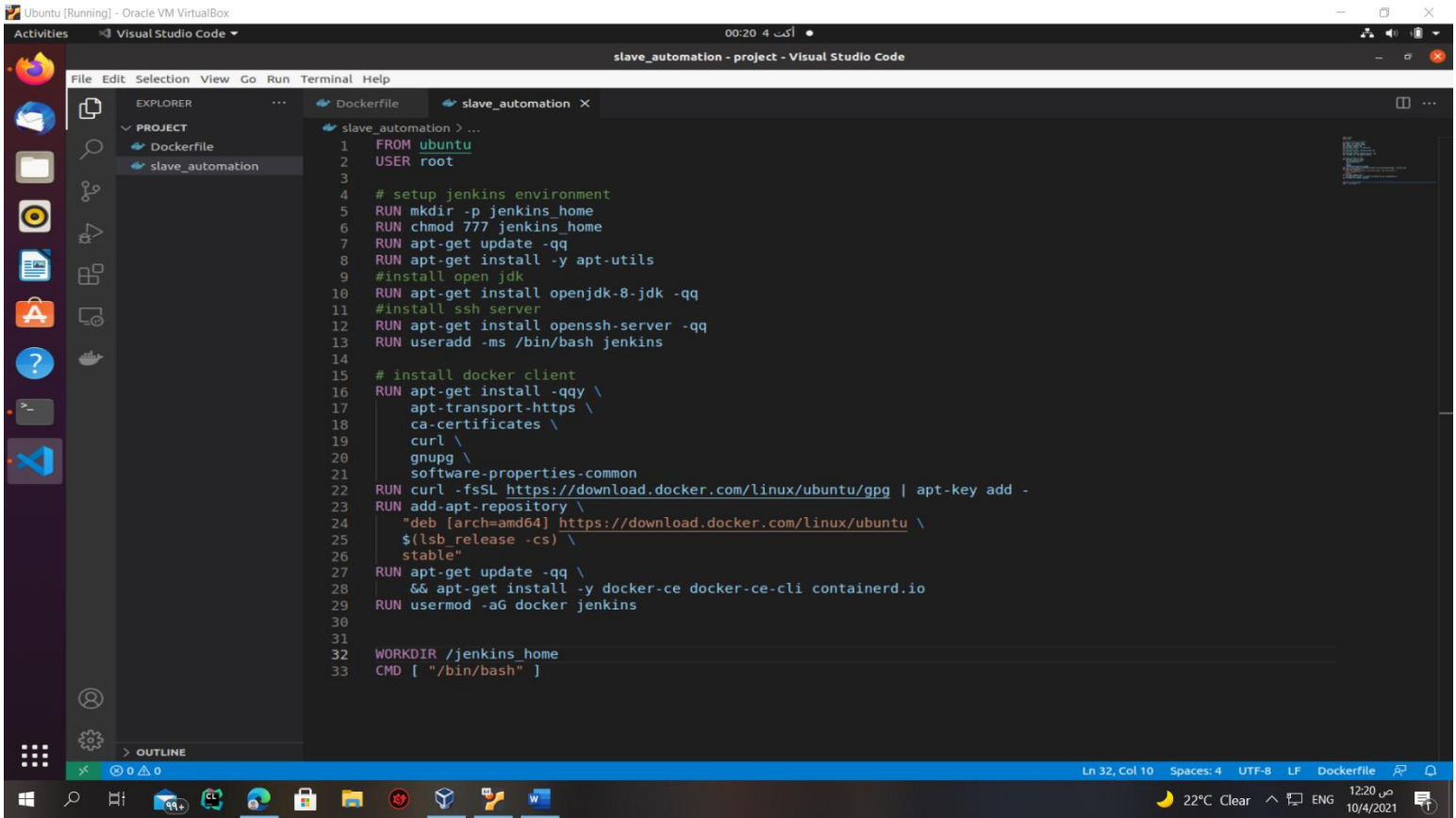


II) Create A multibranch Pipeline

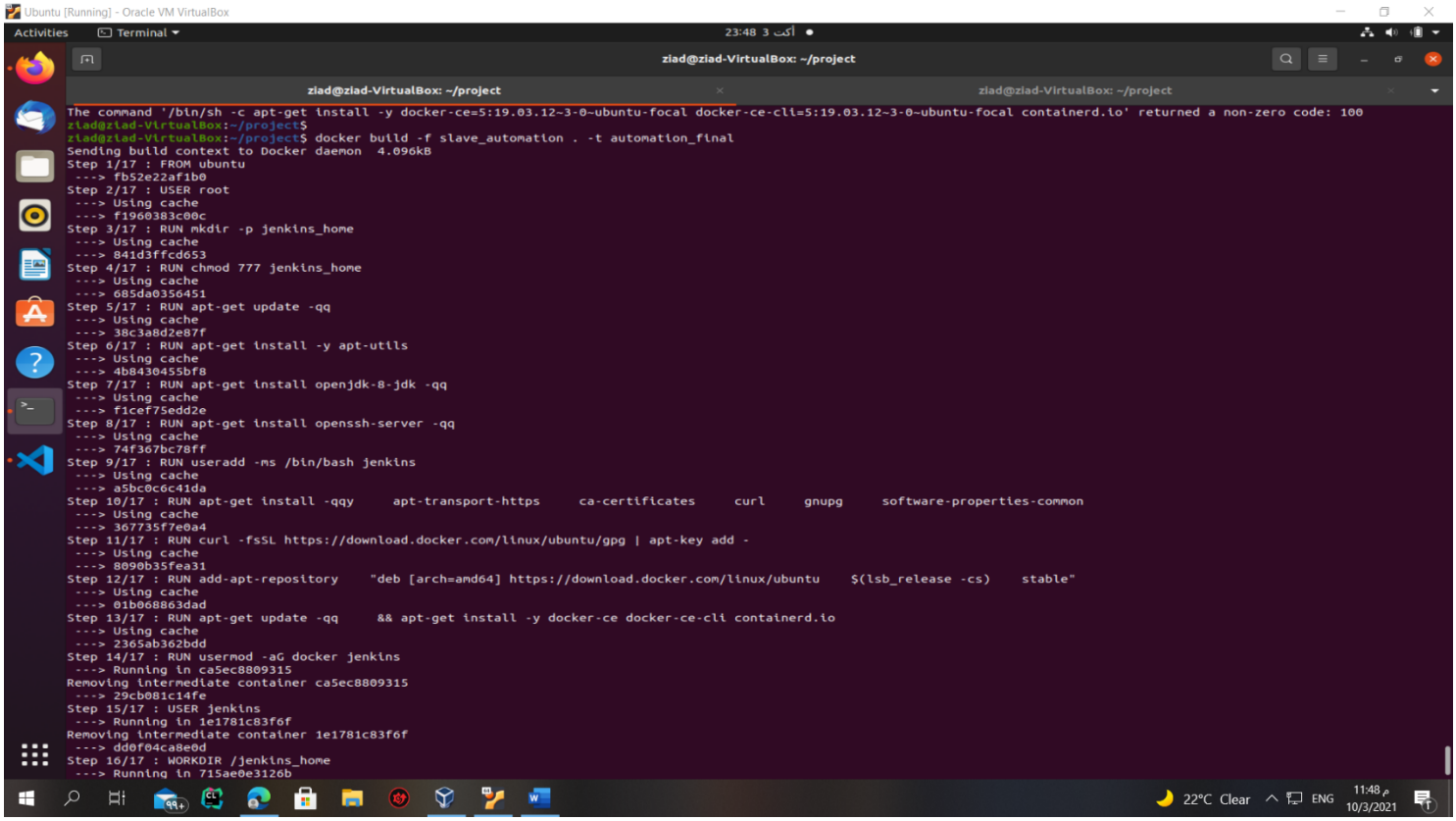


5- Now, what we need is to automate the process of setting up the Jenkins environment, and add the docker client in the slave to be able to use docker, rather than manually setting up the Jenkins environment.

a) Create a docker image than sets up the Jenkins environment and adds the docker client.

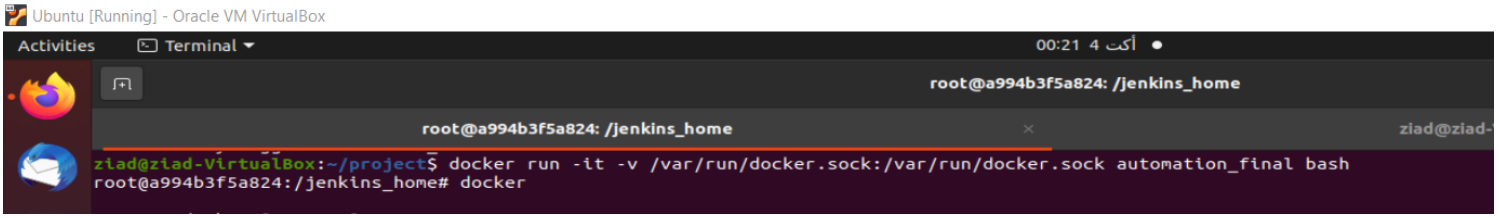


b) Build docker image

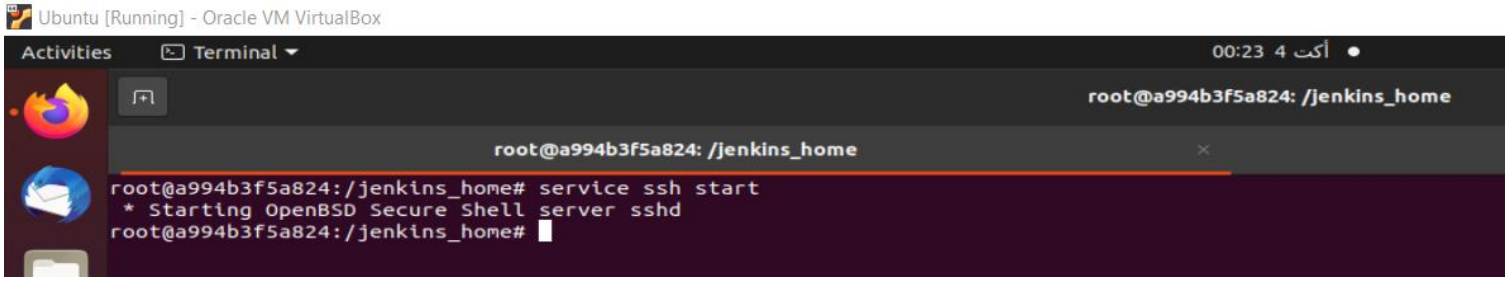


c) Now run the docker container, and bind mount the /var/run/docker.sock file to be able to use docker

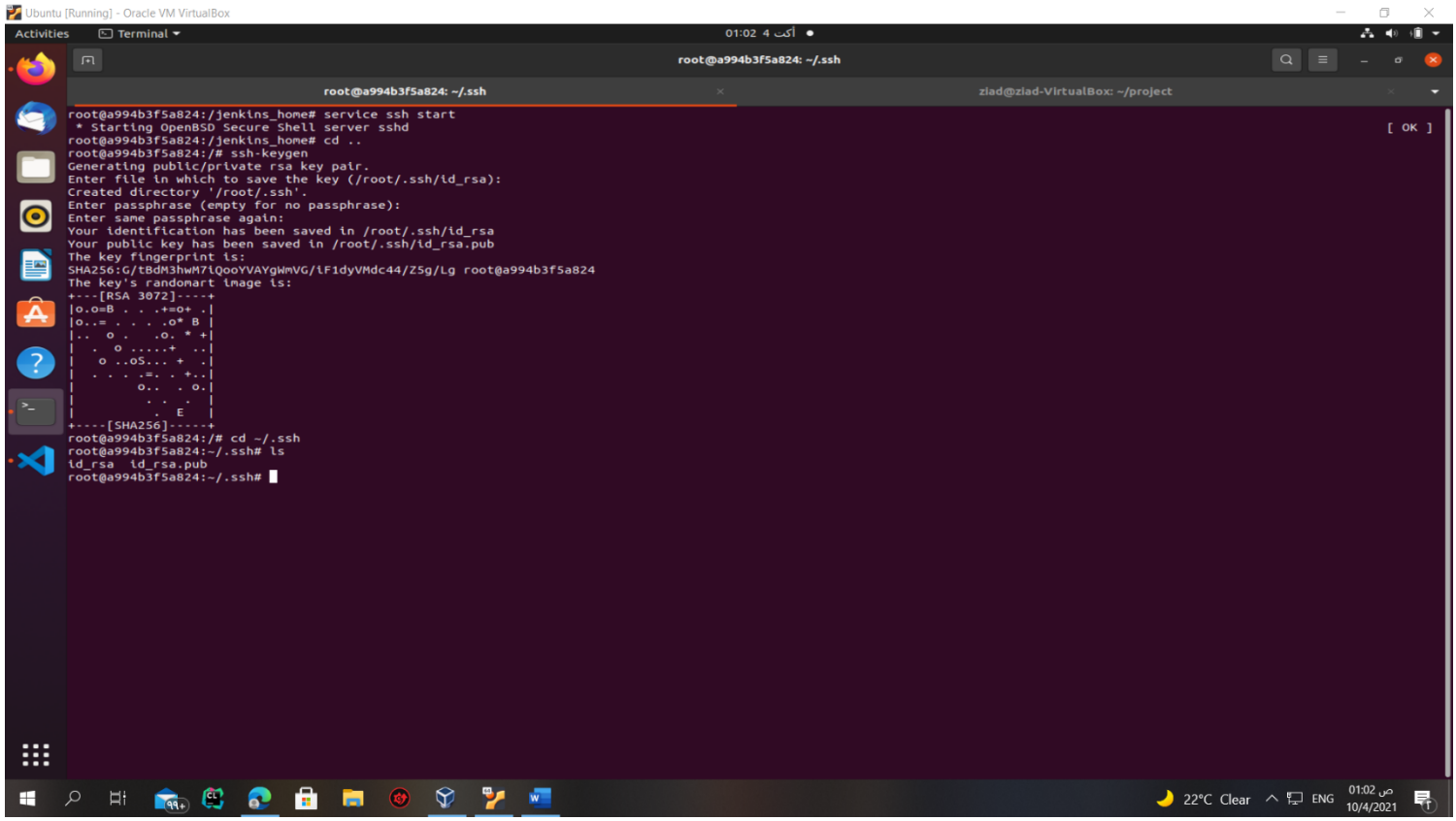
outside of docker rather than docker inside of docker, which is a bad practice.



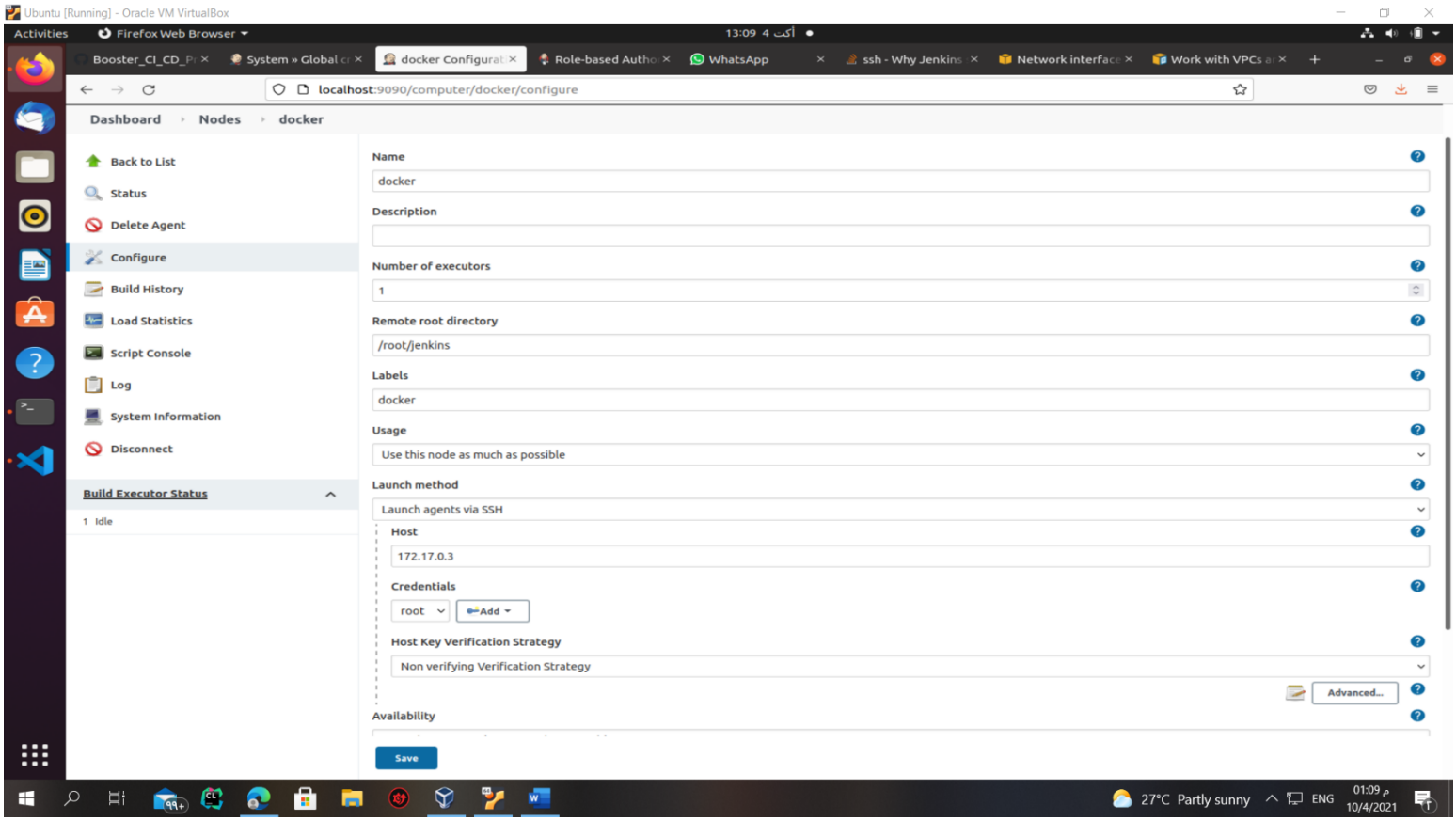
e) start ssh



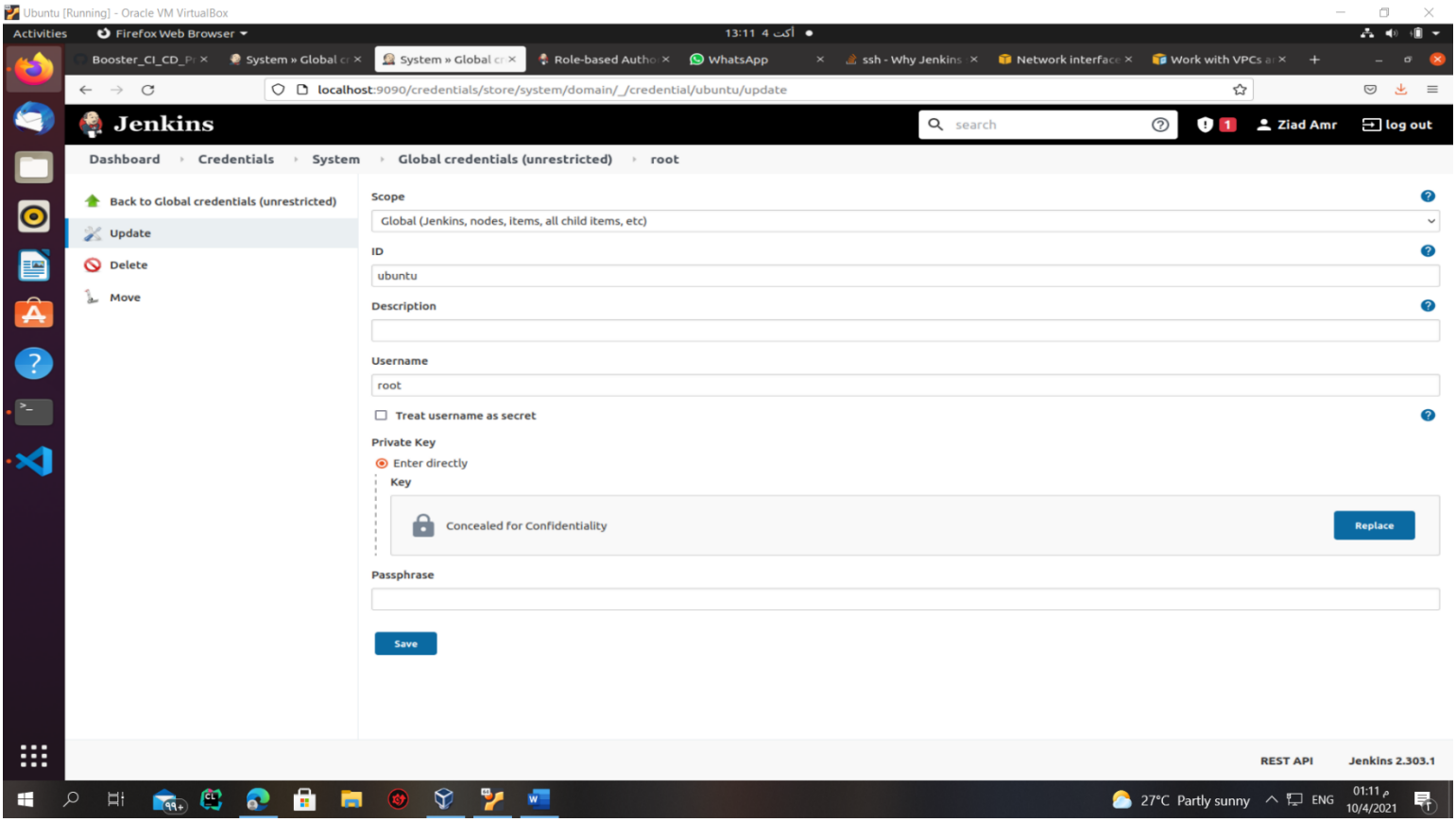
f) generate ssh-keygen, and add public key in authorized_keys file



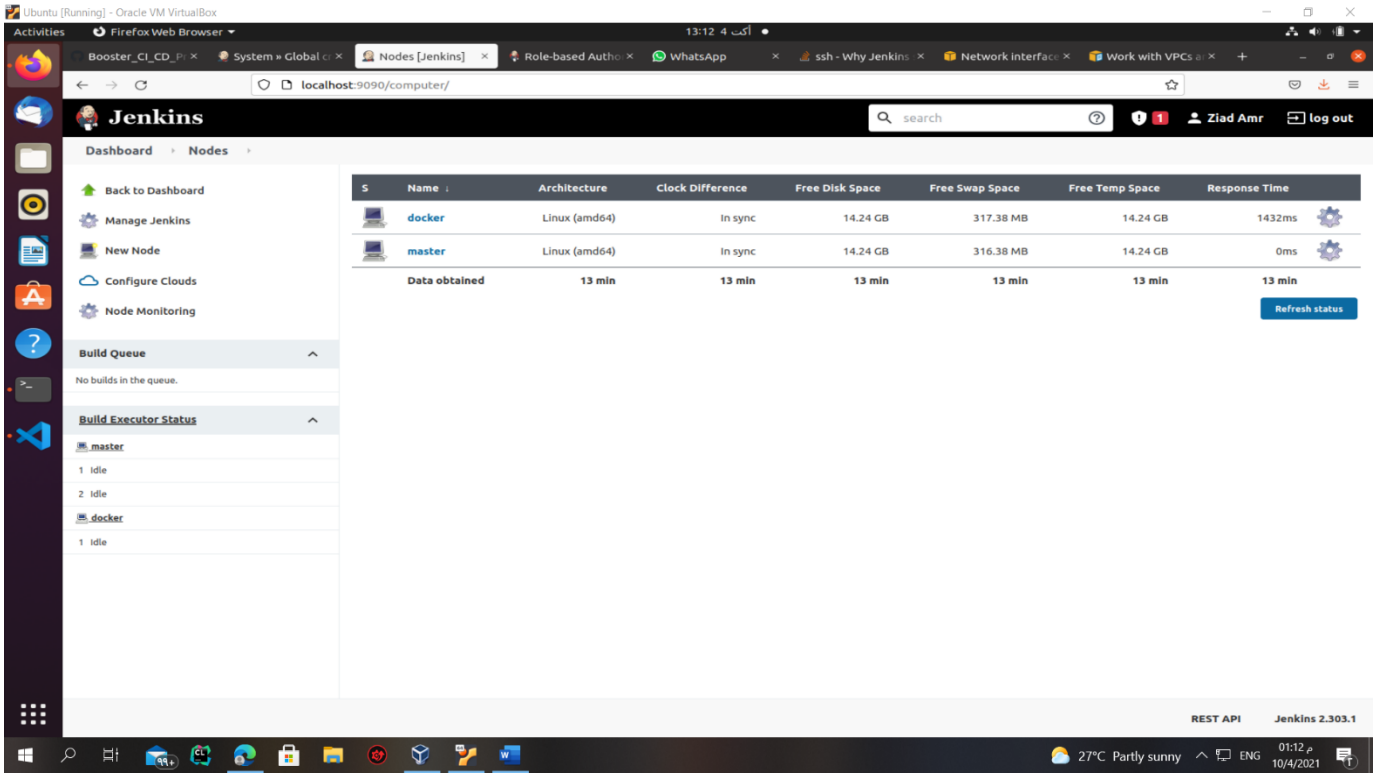
g) now create a node, and configure it, the host should be the ip address of the docker container gotten by docker inspect command



h) Add a credential, and add inside it the public key

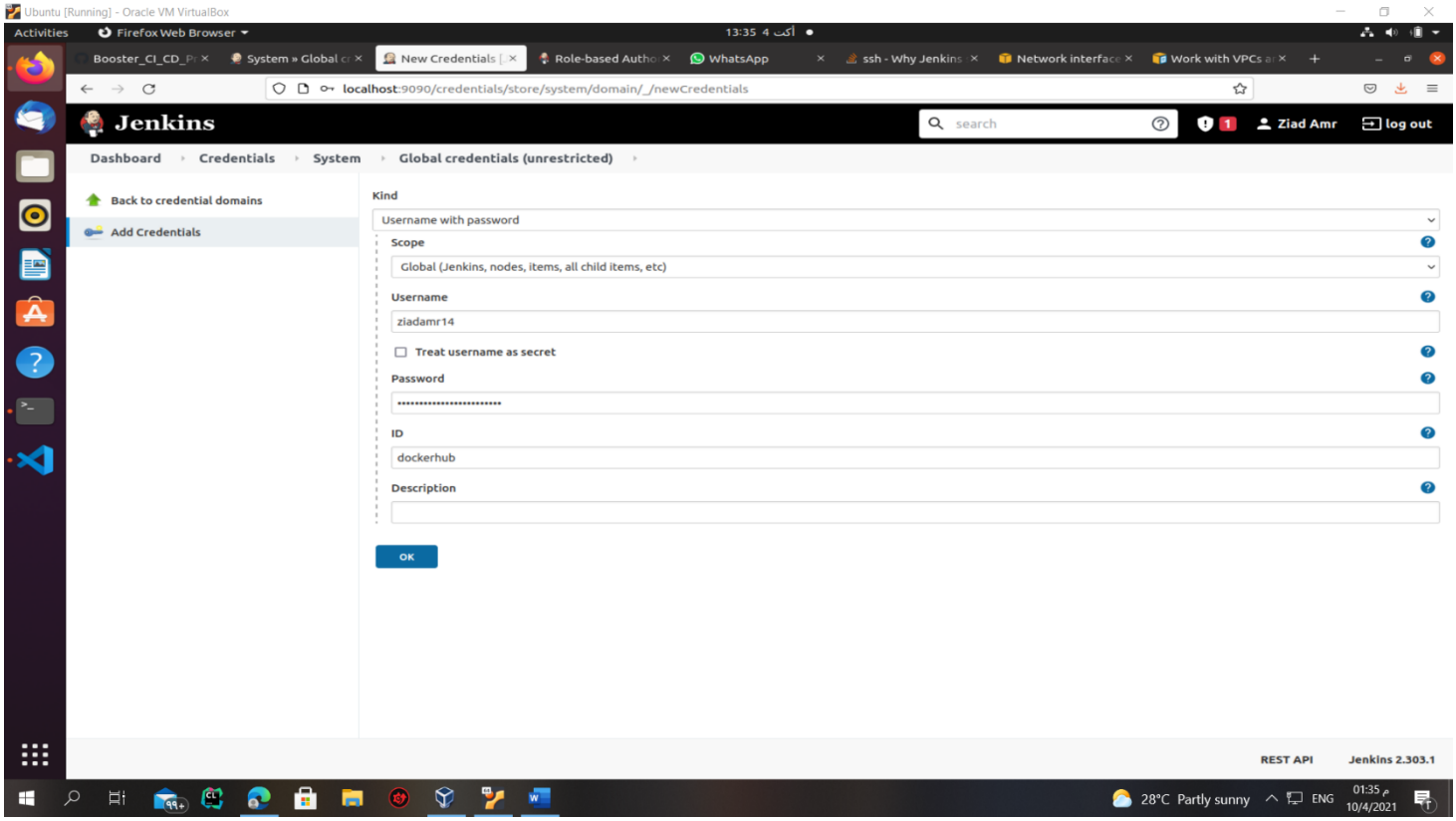


i)Check that node is successfully authenticated

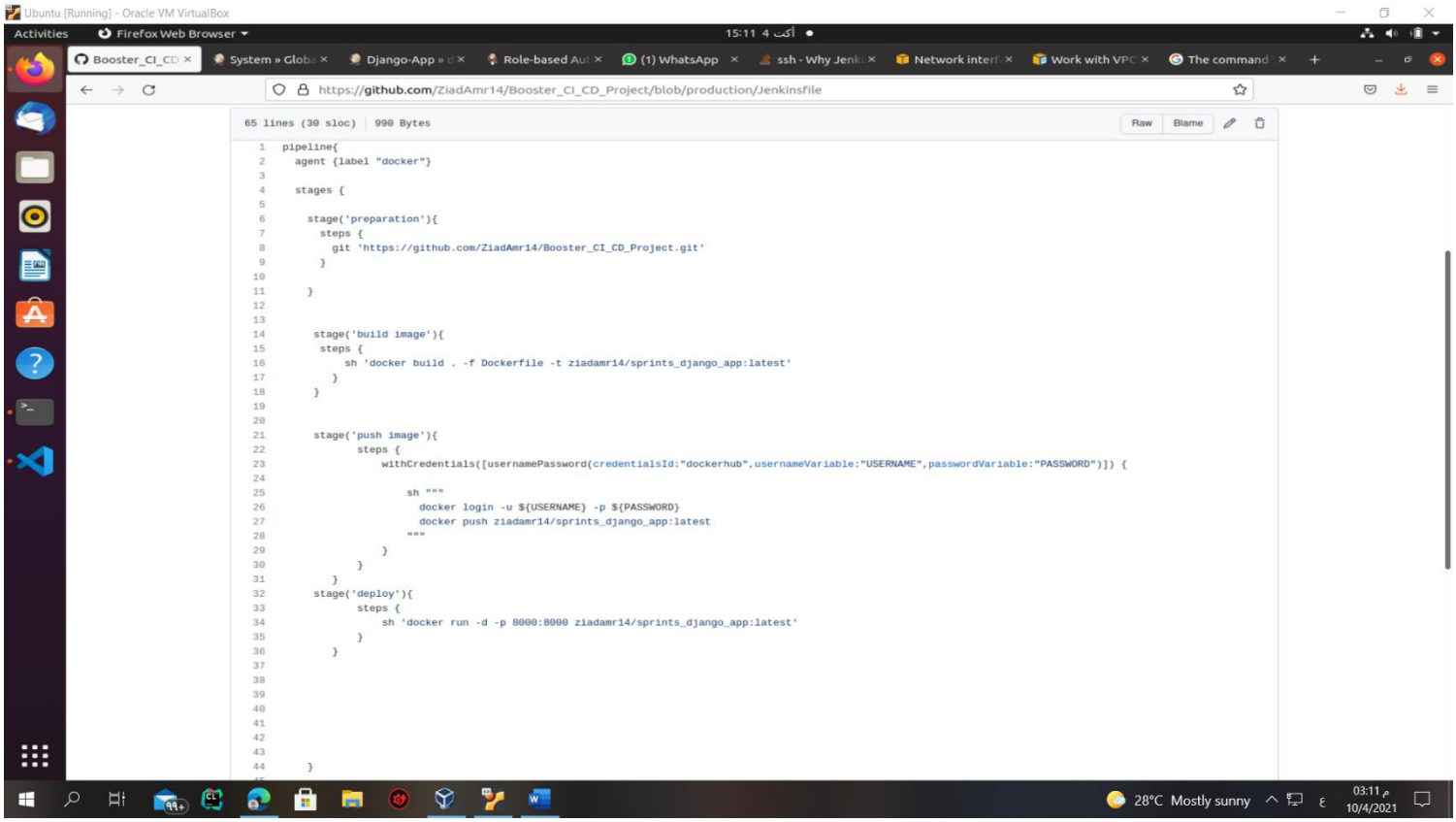


6)Now that everything is configured, we need to write the Jenkins file, we will split the stages into preparation stage, build stage, push stage, deploy stage, and notification stage when we add the slack plugin

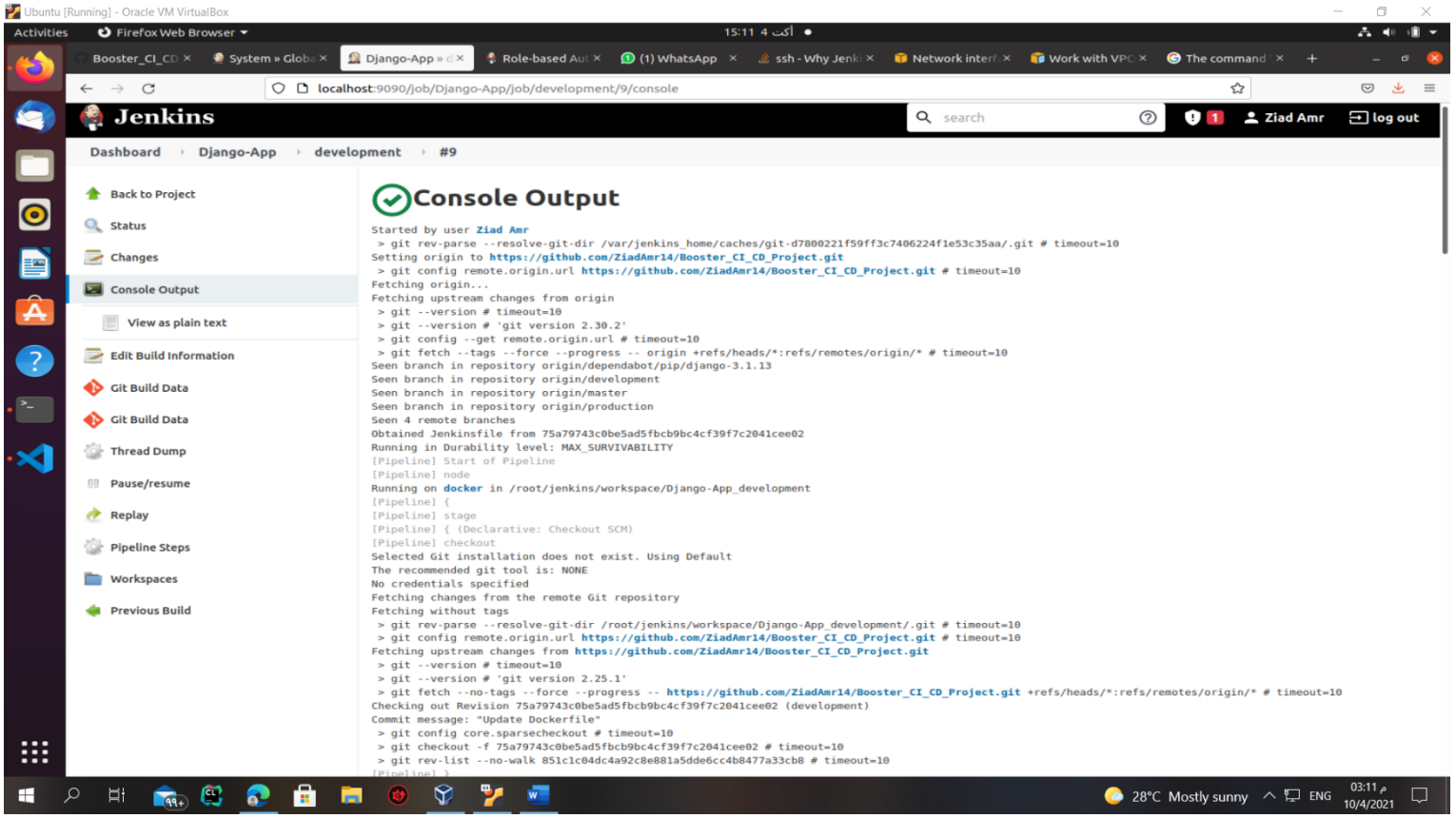
a) first, we need to add a credential for the docker hub username and password.



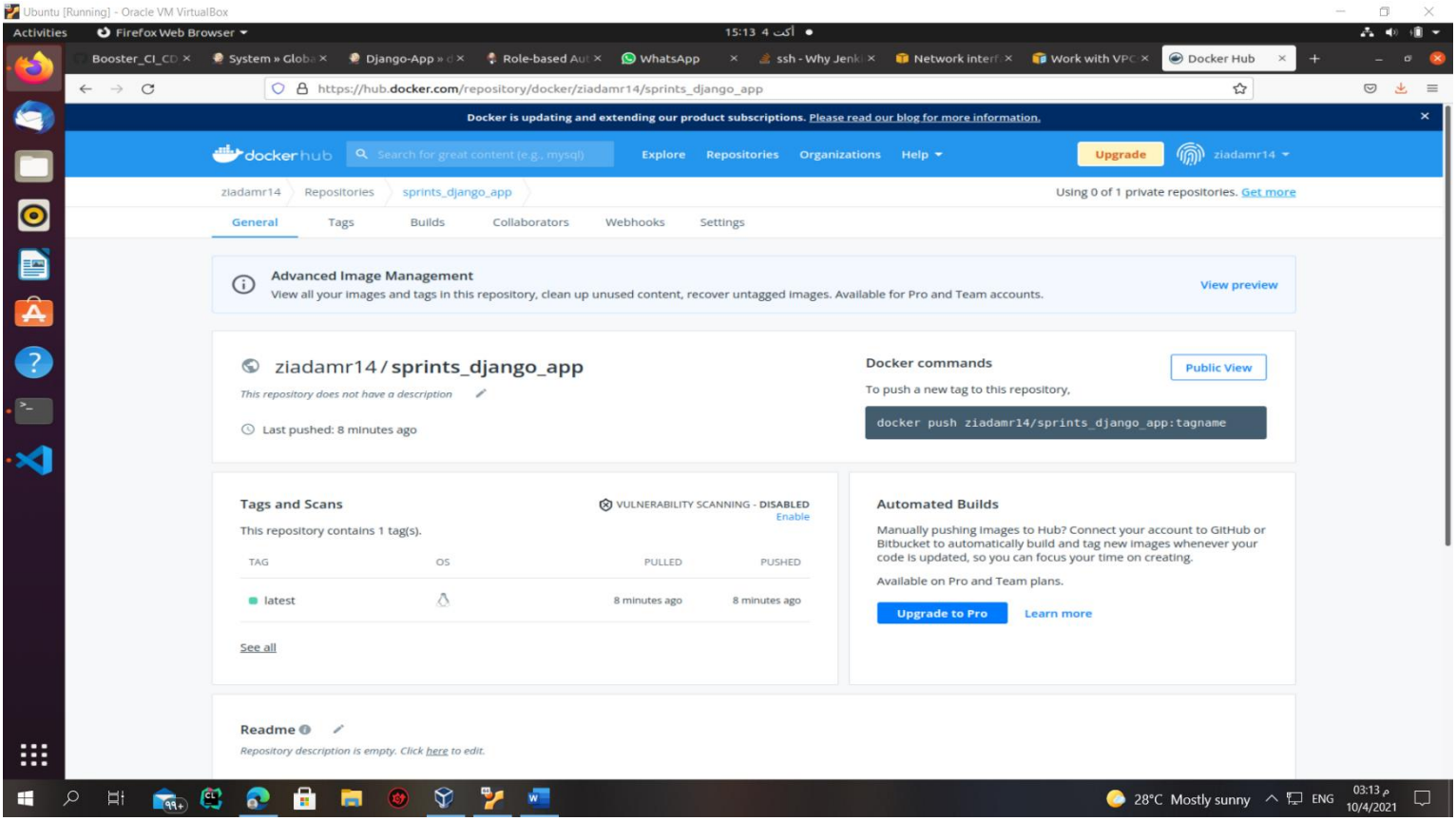
b) Now here is the Jenkins file



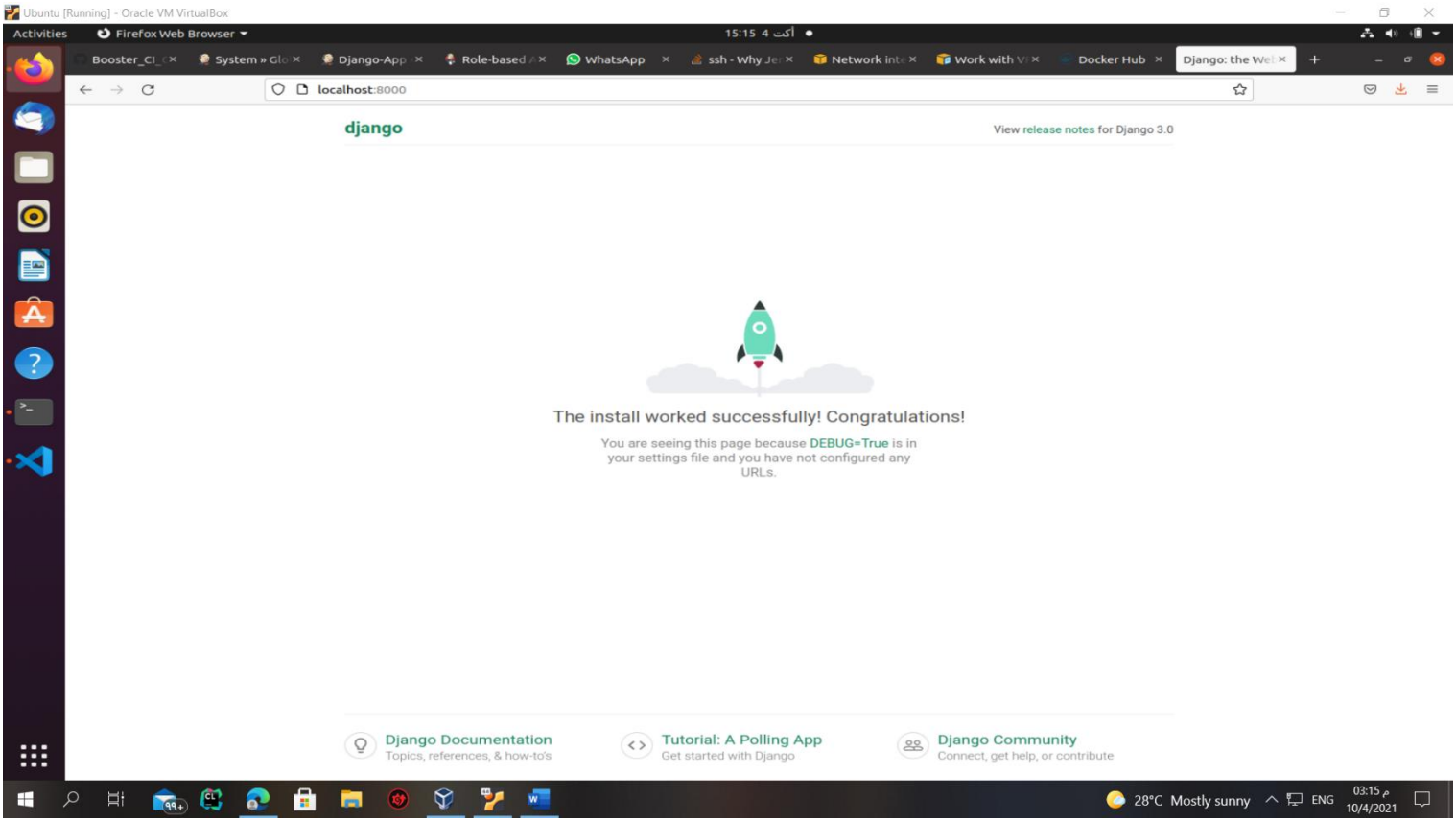
c)Build pipeline



d) Here is the image on dockerhub

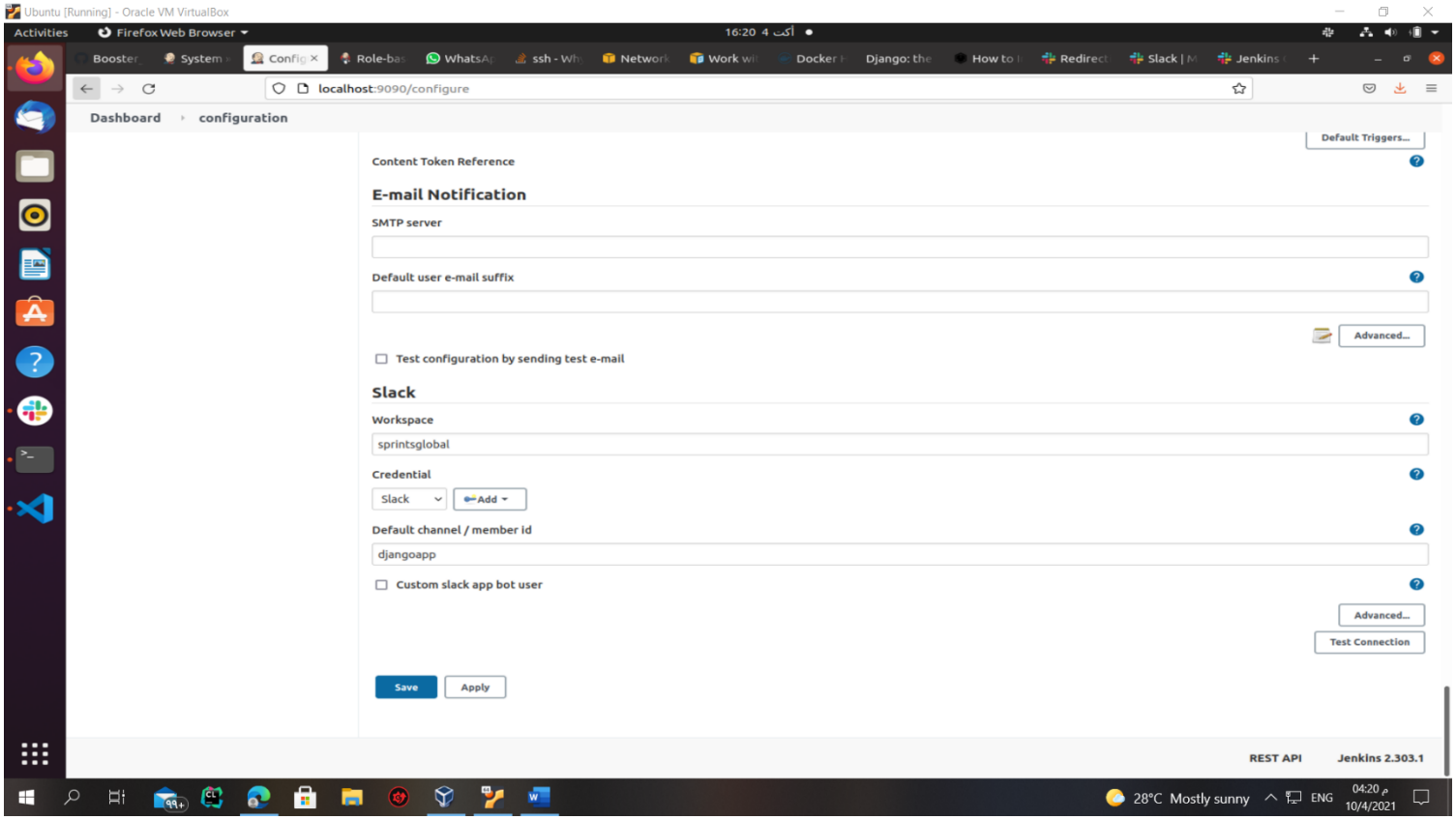


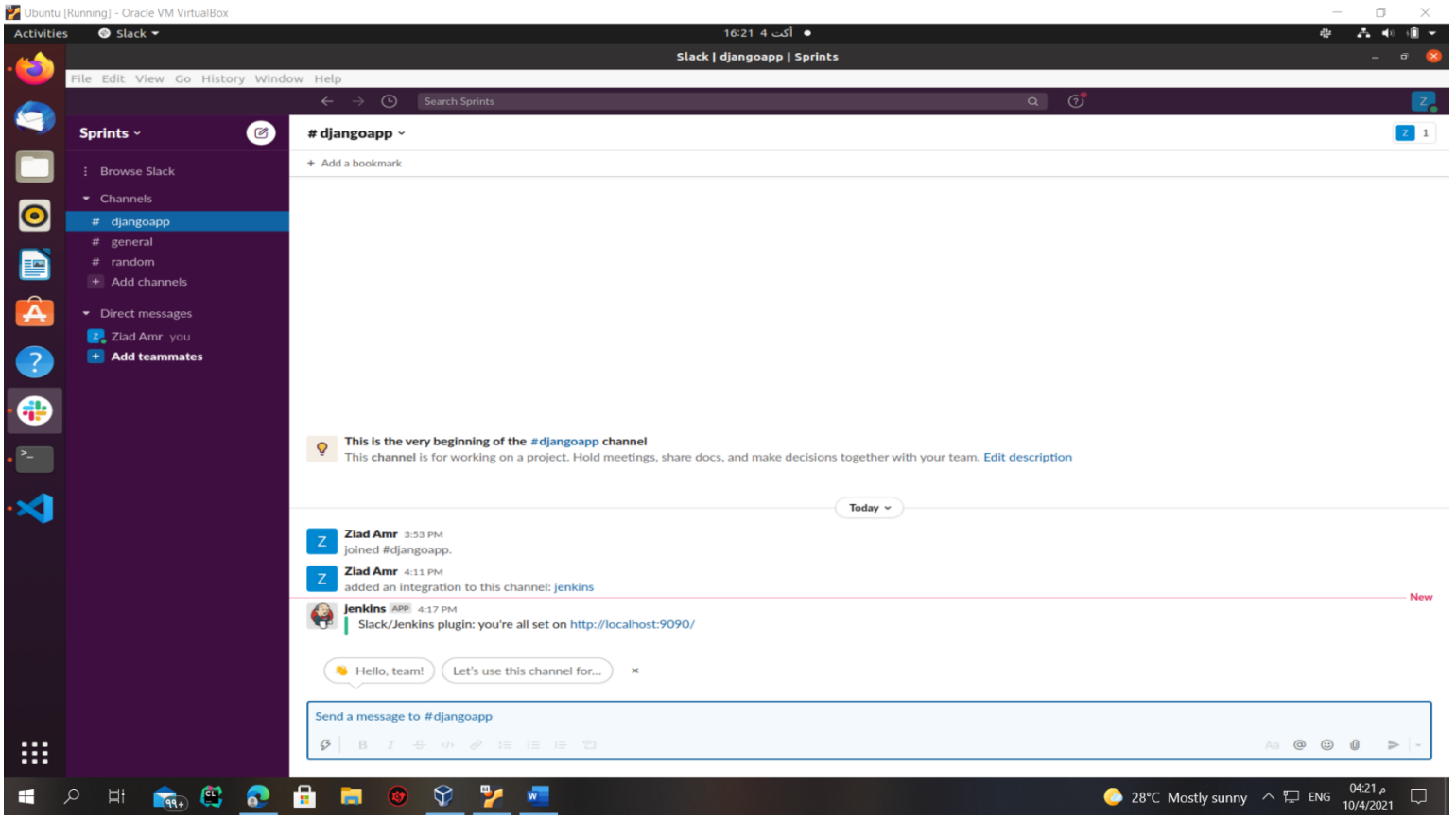
7) Now that the image is uploaded on docker hub, we can access the website through localhost:8000



8) Now, what is remaining is to add the slack plugin

a) Add slack plugin





b) Now let’s add Slack notification in the Jenkins file